CMPSC 464: *Intro to Theory of Computation*
Pennsylvania State University

Professor Sofya Raskhodnikova
March 31, 2016

Homework 9 – Due Thursday, April 7, 2016 before the lecture

Please refer to the general information handout for the full homework policy and options. This homework contains 3 problems, worth 10 points each. *Your solution to each problem should be handed in on a separate sheet of paper.*

Reminder Collaboration is permitted, but you must write the solutions *by yourself without assistance*, and be ready to explain them orally to the instructor if asked. You must also identify your collaborators. Getting solutions from outside sources such as the Web or students not enrolled in the class is strictly forbidden.

Exercises Please practice on exercises 6.1-6.2, 7.1-7.11 and the following exercise.

1. (**Review of asymptotic notation**) For each of the following, answer *true* or *false*. No justification is required.

   (a) $2^{10} = O(n)$

   (b) $16n = O(n)$

   (c) $n^4 = O(n^2 \log n)$

   (d) $n \log n + 10n = O(n^2)$

   (e) $3^n = O(2^n)$

   (f) $3^n = 2^{O(n)}$

   (g) $2^{2^n} = O(2^{2n})$

   (h) $n^n = O(n!)$

   (i) $n = o(n)$

   (j) $2n = o(n^2)$

   (k) $2^n = o(3^n)$

   (l) $1 = o(n)$

   (m) $2 \log n = o(\log n)$

   (n) $\frac{1}{3} = o(1)$

   (o) $\log_2 n = \Theta(\log_3 n)$

   (p) $2^n = \Theta(4^n)$

   (q) $n^5 = \Theta(32^{\log_2 n})$

   (r) $n^3 = \Omega(n^3)$

   (s) $\log n = \Omega(\log(\log n))$

   (t) $2^{5^n} = \Omega(5^{2^n})$

Problems

1. (**Applications of recursion theorem**)

   (a) Let $\text{SMALL}_{\mathsf{TM}} = \{\langle M \rangle \mid M$ is a TM and there is no TM M' equivalent to M which has a much shorter description, that is, $|\langle M' \rangle| \leq \frac{1}{2}|\langle M \rangle|\}$. Show that every infinite subset of $\text{SMALL}_{\mathsf{TM}}$ is not Turing-recognizable.

   (b) Use recursion theorem to give an alternative proof that ALL_{TM} is not Turing-recognizable. Recall that $ALL_{\mathsf{TM}} = \{\langle M \rangle \mid M$ is a TM and $L(M) = \Sigma^*$, where Σ is the input alphabet of $M\}$.

2. (**Exponentiation cipher**) An exponentiation cipher encodes a message A using a ciphertext $C = A^e \pmod{p}$ where p is a prime number and e is an integer exponent. (Here A and C are also integers.) You are given integers A, C, e and p, and you would like to determine whether C is a valid ciphertext for message A.

   (a) Formulate this problem as a language EC.

   (b) Explain why the following algorithm for EC does not run in polynomial time: *Compute A^e using $e - 1$ multiplications. Take the result modulo p using one integer division, and compare the answer to C.*

   (c) Show that $EC \in \mathrm{P}$. Analyze the running time of your algorithm using O-notation.
   Hint: First, find an algorithm for the case when e is a power of 2.

3. (**Closure properties of P**) For both parts of this problem, analyze the running time of your algorithms using O-notation. Prove that P is closed under

   (a) (**3 points**) concatenation;

   (b) (**7 points**) star.
   Hint: Use dynamic programming. On input $y = y_1 \cdots y_n$ for $y_i \in \Sigma$, build a table indicating for each $i \leq j$ whether the substring $y_i \cdots y_j \in A^*$ for any $A \in \mathrm{P}$.

   Think about union and complement on your own.